The Path to Becoming a Software Architect

Translation: 成为软件架构师的途径

Image source: monument valley game

Translation: 图片来源：纪念碑谷游戏

Have you ever wondered what career opportunities a developer has? What directions are open, beyond what horizons to grow. And most importantly, where are developers beyond the age of 45? Is there a developer among your friends who is over 45? I know several developers beyond this age, and many of them are hardcore programmers who even saw punch cards back in the day.

Translation: 你有没有想过开发人员有什么职业机会？什么方向是开放的，超越什么地平线才能成长。最重要的是，超过45岁的开发商在哪里？你的朋友中有没有超过45岁的开发人员？我认识几个超过这个年龄的开发人员，他们中的许多人都是铁杆程序员，他们甚至在过去看到过穿孔卡片。

There are several career paths a developer might take:

Translation: 开发人员可能会走以下几种职业道路：

● The first and obvious one is to grow in the area in which you are working. If you are a junior developer, then become to middle, then senior and lead roles.

Translation: ● 第一个也是显而易见的是在你工作的领域成长。如果你是一个初级开发人员，那么你可以成为中级、高级和领导角色。

● Transition to another technology stack. A significant number of developers moved into the mobile area when iOS and Android OS gained ground.

Translation: ● 过渡到另一个技术堆栈。当iOS和Android操作系统站稳脚跟时，大量开发者进入了移动领域。

● Grow into a manager role. As a developer, the hugest staffing issue I saw was the shortage of competent managers. Smart managers are expensive, hence they are scarce. If the manager has a technical background, that will allow him to be on the same wavelength with the developers.

Translation: ● 成长为管理者。作为一名开发人员，我看到的最大的人员配置问题是缺乏称职的管理人员。聪明的管理者是昂贵的，因此他们是稀缺的。如果经理有技术背景，这将使他能够与开发人员保持一致。

● Become a software architect. This direction will be considered in this series of articles.

Translation: ● 成为一名软件架构师。这一方向将在本系列文章中加以考虑。

● Get out of IT. Sometimes this happens. It is never too late to do what you like to do.

Translation: ● 离开IT。有时会发生这种情况。做你喜欢做的事永远不会太迟。

Article series

Translation: 文章系列

The Path to Becoming a Software Architect

Translation: 成为软件架构师的途径

Stakeholders in Software Architecture

Translation: 软件体系结构中的利益相关者

Types of Software Architects

Translation: 软件架构师的类型

Quality attributes in Software Architecture

Translation: 软件体系结构中的质量属性

Documentation in Software Architecture

Translation: 软件体系结构中的文档

Certificates in Software Architecture

Translation: 软件体系结构证书

Books in Software Architecture

Translation: 软件体系结构书籍

System Design Cheat Sheet

Translation: 系统设计备忘单

Yes, This Was My Path

Translation: 是的，这是我的路

In the past eight years, I have worked with Java EE, then moved to iOS, and became a team lead. I managed various developer teams, including Android, and Web stacks. Created the architecture of the network layer for several services developed by the company, with sockets and REST API. I became acquainted with the managerial role and the prospect of growing in this direction while in the position of team lead for over two years. In my next role, my goal is to grow as a software architect.

Translation: 在过去的八年里，我一直在Java EE工作，后来转到iOS，并成为团队负责人。我管理过各种开发团队，包括Android和Web堆栈。为该公司开发的几个服务创建了网络层的架构，包括套接字和RESTneneneba API。在担任团队领导两年多的时间里，我开始熟悉管理角色以及朝着这个方向发展的前景。在我的下一个角色中，我的目标是成长为一名软件架构师。

For most developers, the function of the architect on the project is often unclear, so in this series of articles, I will try to find the answers to these related questions. Who is an architect, what is the scope of responsibilities, and how to grow in this direction and outline an action plan for myself and beginners wanting to move along this path?

Translation: 对于大多数开发人员来说，架构师在项目中的作用往往是不明确的，因此在本系列文章中，我将尝试找到这些相关问题的答案。谁是建筑师，职责范围是什么，如何朝着这个方向发展，并为我自己和想要沿着这条道路前进的初学者制定行动计划？

Who Can Benefit from This?

Translation: 谁能从中受益？

This series of articles will help you if you belong to one of the following categories:

Translation: 如果您属于以下类别之一，本系列文章将对您有所帮助：

● IT developer or engineer. You are still growing as a developer, but you are looking ahead and planning your career. Even if the goals are initially vague, a person who consciously sets strategic goals will reach them much quicker than a person who does not plan where she is heading.

Translation: ● IT开发人员或工程师。作为一名开发人员，你仍在成长，但你正在展望未来，规划自己的职业生涯。即使目标最初是模糊的，一个有意识地设定战略目标的人也会比一个没有计划前进方向的人更快地实现目标。

● Team leader, lead software engineer. You are at the highest stage of the software development discipline. To grow further, you have a choice to either learn one more stack of technologies, pursue a career outside software engineering, or to become a software architect.

Translation: ● 组长，首席软件工程师。您正处于软件开发学科的最高阶段。为了进一步发展，你可以选择再学习一堆技术，从事软件工程以外的职业，或者成为一名软件架构师。

● Software architect. You recently took this position, or have been working in this field for a long time. Perhaps one of the main qualities of such a specialist is the understanding that there are always areas that a person does not know and that the learning process is continuous.

Translation: ● 软件架构师。你最近担任了这个职位，或者已经在这个领域工作了很长时间。也许这样一位专家的主要品质之一是理解总有一些领域是一个人不知道的，而且学习过程是连续的。

● IT manager. Although you are a manager, you understand perfectly well that you should at least approximately understand what your subordinates or colleagues are doing. The acute problem of management is the technical incompetence of the manager in the field in which he or she is managing.

Translation: ● IT经理。尽管你是一名经理，但你非常清楚，你至少应该大致了解你的下属或同事在做什么。管理的尖锐问题是经理在他或她所管理的领域技术上的无能。

Who is an Architect?

Translation: 谁是建筑师？

Before moving on to more specific questions, it is necessary to define the software architect role and responsibilities.

Translation: 在讨论更具体的问题之前，有必要定义软件架构师的角色和职责。

A software architect is a software expert who makes high-level design choices and dictates technical standards, including software coding standards, tools, and platforms. The leading expert

is referred to as the chief architect. (Wikipedia, The Free Encyclopedia, s.v. "Software architect", https://en.wikipedia.org/wiki/Software_architect

Translation: 软件架构师是指做出高级设计选择并规定技术标准（包括软件编码标准、工具和平台）的软件专家。首席专家被称为首席建筑师。（维基百科，自由百科全书，s.v."软件架构师"，https://en.wikipedia.org/wiki/Software_architect

Like most high-level positions, there are no clear criteria that define this role. However, it is possible to identify several responsibilities and qualities that contribute to the career of the architect.

Translation: 与大多数高级职位一样，没有明确的标准来定义这一角色。然而，可以确定对建筑师的职业生涯有贡献的几个责任和素质。

First, let's consider the characteristics of the architect:

Translation: 首先，让我们考虑一下建筑师的特点：

● Communicability. Having talked with many software architects, I heard that it is one of the essential characteristics of this specialist. During the working day, they have to speak with customers in the language of business, managers of all levels, business analysts, and developers. If you have a natural charisma and you know how to convince people, then this will be a huge plus, as it is crucial to explain your actions correctly. Architects are laconic, eloquent, and competent speakers. The software architects with whom I spoke have highly developed skills in communication and persuasion. Another reason why this characteristic is most important is that the architect in this role participates in most discussion making processes, and often compromises must be reached that is acceptable and beneficial for all involved parties.

Translation: ● 传染性。在与许多软件架构师交谈后，我听说这是这位专家的基本特征之一。在工作日，他们必须用业务语言与客户、各级经理、业务分析师和开发人员交谈。如果你有天生的魅力，并且你知道如何说服别人，那么这将是一个巨大的优势，因为正确解释你的行为至关重要。建筑师简洁、能说会道、能言善辩。与我交谈过的软件架构师在沟通和说服方面有着高度发展的技能。这一特性最重要的另一个原因是，担任这一角色的架构师参与了大多数讨论过程，通常必须达成对所有相关方都可接受和有益的妥协。

● Broad and deep technical knowledge. It should be obvious since one cannot become a software architect with a medical background. Besides, the architect usually has expertise in several technological stacks at a decent level and should have a good understanding of a few other ones. The software architect should also be prepared to compose a large number of technical documentation, reports, and diagrams.

Translation: ● 广泛而深入的技术知识。这应该是显而易见的，因为一个人不能成为一个有医学背景的软件架构师。此外，架构师通常在几个技术堆栈中具有相当水平的专业知识，并且应该对其他一些堆栈有很好的理解。软件架构师还应该准备好编写大量的技术文档、报告和图表。

● Responsibility. You should understand that architect decisions are usually the most expensive. Therefore, a person in this position should take the most responsible approach to his work and the decisions made. If the developer's error costs a couple of days of work of one person, then the architect's mistake can cost person-years on complex projects!

Translation: ● 责任您应该明白，架构师的决策通常是最昂贵的。因此，担任这一职务的人应该对自己的工作和所做的决定采取最负责任的态度。如果开发人员的错误花费了一个人几天的工作，那么建筑师的错误可能会在复杂的项目上花费人年的时间！

● Stress resistance. You will have to make decisions because in this role, you will be asked to do so, and you will need a response. You will be working with different people from different areas, and you will have to deal with rapidly changing demands or even with changing business environments. Therefore, it is necessary to be ready for stress and to look for some ways to escape negative emotions. Work is always more pleasant when it brings pleasure. So if you choose this role only for the money, then think again.

Translation: ● 抗压性。你必须做出决定，因为在这个角色中，你会被要求这样做，你需要回应。你将与来自不同领域的不同人员一起工作，你将不得不应对快速变化的需求，甚至是不断变化的商业环境。因此，有必要为压力做好准备，并寻找一些逃避负面情绪的方法。当工作带来快乐时，它总是更令人愉快。所以，如果你选择这个角色只是为了钱，那就三思而后行。

● Management skills. This includes both organizational and leadership skills. The ability to lead a team, which may be distributed and composed of very different specialists, is essential.

Translation: ● 管理技能。这包括组织能力和领导能力。领导一个团队的能力是至关重要的，这个团队可能是分散的，由非常不同的专家组成。

● Analytic skills. Even if a specialist has a broad erudition in technology, he has tried many things on his own or participated in projects of various types. It does not guarantee that he can easily change the style of thinking to an architect. One of the most valuable tasks is the ability to represent an abstract problem in the form of some finite real object of the system, which developers are already evaluating, designing, and developing.

Excellent communication skills are essential to represent the abstraction in the form of the final system to the members of the team and the customer. It will be necessary to communicate with both business and development, which is still to be done.

Translation: ● 分析能力。即使一个专家在技术方面有着广泛的学识，他也会自己尝试很多事情，或者参与各种类型的项目。这并不能保证他能轻易地将思维方式转变为建筑师。最有价值的任务之一是将抽象问题表示为系统的某些有限实物的能力，开发人员已经在评估、设计和开发这些实物。优秀的沟通技巧对于以最终系统的形式向团队成员和客户表示抽象至关重要。有必要与业务和开发部门进行沟通，这仍有待完成。

If we talk about the responsibilities of the architect, then here is the perfect example from the 19th century about bridge construction. At that time, the tests of the newly constructed bridge were the following: the key group of engineers, architects and workers stood under the bridge while the first vehicles were on it. Thus, they staked their lives upon the construction and the strength of the structure. So if there is a question — what is the responsibility of the software architect on the project? The answer is, he is responsible for everything.

Translation: 如果我们谈论建筑师的责任，那么这里是19世纪关于桥梁建设的完美例子。当时，新建桥梁的测试如下：当第一批车辆在桥上时，关键的工程师、建筑师和工人站在桥下。因此，他们将生命押在了结构的施工和强度上。所以，如果有一个问题——软件架构师对项目的责任是什么？答案是，他对一切负责。

If you give up loud and beautiful phrases, then the architect's work includes:

Translation: 如果你放弃了响亮而优美的短语，那么建筑师的工作包括：

● Identifying the stakeholders on the project.

Translation: ● 确定项目的利益相关者。

● Identifying business requirements and requirements of the stakeholders on the project.

Translation: ● 确定业务需求和利益相关者对项目的要求。

● Designing the entire system based on the received requirements.

Translation: ● 根据收到的要求设计整个系统。

● Choosing the system architecture and each component of this system at a high level.

Translation: ● 在高层次上选择系统架构和该系统的每个组件。

● Choosing the technologies for the implementation of each component and connections between the parts.

Translation: ● *选择用于实现每个组件的技术以及部件之间的连接。*

● Architectural review. Yes, yes, it exists.

Translation: ● *建筑审查。是的，是的，它是存在的。*

● Code-review.

Translation: ● *代码审查。*

● Writing project documentation and support it.

Translation: ● *编写项目文档并为其提供支持。*

● Creating unified development standards in the company.

Translation: ● *在公司内创建统一的开发标准。*

● Controlling the architecture during the next iteration of the system release.

Translation: ● *在系统发布的下一次迭代中控制体系结构。*

It is only a subset of the software architect's responsibilities. The most important responsibility is full technical support of the project from the moment of inception, through product release, to the development of enhancements. And supporting the next releases. It will be necessary to switch a lot between different tasks during the working day.

Translation: 这只是软件架构师职责的一个子集。最重要的责任是从项目开始的那一刻起，通过产品发布，到增强功能的开发，为项目提供全面的技术支持。并支持下一个版本。在工作日中，有必要在不同的任务之间进行大量切换。

How to Become a Software Architect?

Translation: 如何成为一名软件架构师？

To begin with, it is requisite to define milestone goals that lead to achieving your strategic goal of becoming a software architect. For me, such goals for the next six months are:

Translation: 首先，必须定义里程碑式的目标，以实现成为软件架构师的战略目标。对我来说，未来六个月的目标是：

● Understand and try several technological stacks. My current knowledge is concentrated in the field of iOS. It is necessary to try Android, several server languages, to start python, and refresh Java EE skills. The architect is a full-stack developer, so it is essential to have a broad technical knowledge.

Translation: ● 了解并尝试几种技术堆栈。我目前的知识主要集中在iOS领域。有必要尝试Android，几种服务器语言，以启动python，并刷新Java EE技能。架构师是一个全栈开发人员，因此拥有广泛的技术知识是至关重要的。

● Reading literature. It is required to determine the most valuable books and articles that will help to grow in this direction. Usually, the most effective way to find such literature is to ask other professionals in this field for their recommendation. In one of the future articles, I plan to give you such a list of literature.

Translation: ● 阅读文学。需要确定有助于朝着这个方向发展的最有价值的书籍和文章。通常，找到此类文献最有效的方法是向该领域的其他专业人士寻求推荐。在未来的一篇文章中，我计划给你一份这样的文献清单。

● Find a mentor. It is desirable to find a software architect at your current place of employment. It is always easier to get experience from a trained specialist than to start considering a particular area from scratch. It is requisite to be prepared to ask the right questions from your mentor.

Translation: ● 找一个导师。最好在你目前的工作地点找到一位软件架构师。从受过训练的专家那里获得经验总是比从头开始考虑某个特定领域更容易。有必要准备好向你的导师提出正确的问题。

● Study courses/obtain certificates. There are many courses and certifications available, but only a few are worth their money, and the higher level courses cost a lot of money. I have attended the architectural courses of Luxoft (http://www.luxoft-training.com/it-course/ARC-001/), which have proven to be a worthy investment. It is crucial that the lecturer of the course be a professional in this field and be able to answer questions. As for certificates, before starting, it is best to understand whether there are authoritative certification systems for architects and whether it is worthwhile obtaining the certification. This point I will discuss in a future article of this series.

Translation: ● 学习课程/获得证书。有很多课程和证书可供选择，但只有少数课程是物有所值的，更高级别的课程要花很多钱。我参加过鲁软的建筑课程(http://www.luxoft-training.com/it-course/ARC-001/)，这已被证明是一项有价值的投资。至关重要的是，该课程的讲师必须是该领域的专业人士，并能够回答问题。至于证书，在开始之前，最好了解

建筑师是否有权威的认证体系，以及是否值得获得证书。这一点我将在本系列的未来文章中讨论。

One of the essential parts is a clear and stable plan review. What has been done, what should be reviewed, and where to accelerate or which goal to remove as useless.

Translation: 其中一个重要部分是明确和稳定的计划审查。已经做了什么，应该审查什么，在哪里加速或取消哪个目标是无用的。

Check Your Readiness Level

Translation: 检查您的准备程度

If you are interested in this introductory article from the series on how to become a software architect, or if you suddenly have thoughts to try this path, it is worth making sure that you want it.

Translation: 如果你对本系列中关于如何成为软件架构师的介绍性文章感兴趣，或者如果你突然有了尝试这条道路的想法，那么值得确保你想要它。

Firstly, people are afraid of everything new. A new position, new kind of stress, as opposed to the comfortable status quo. Of course, the choice is not always unambiguous and depends on how much you are willing to change something in your life. At the same time, it can depend not only on you but also on the family, your financial commitments, parents, and other factors.

Translation: 首先，人们害怕一切新事物。一个新的位置，一种新的压力，与舒适的现状相反。当然，选择并不总是明确的，取决于你愿意在多大程度上改变你的生活。同时，这不仅取决于你，还取决于家庭、你的经济承诺、父母和其他因素。

Secondly, this path takes several years. The process of becoming a software architect does not happen overnight. As a team lead, I realized what to do and how to deal with stress only a year after I was appointed to an official position. At the same time, six months before that, I performed it unofficially. One software architect I know said that he understood what his responsibilities are 18 months after he was promoted to this role. Such intervals of time are reasonable, and you need to know whether you are ready to move in this direction. Even if you do not have a stable plan available, it is better to start taking small steps that move you ahead, rather than remaining in the same place.

Translation: 其次，这条道路需要几年时间。成为软件架构师的过程不是一蹴而就的。作为一名团队领导，我在被任命为正式职位一年后才意识到该做什么以及如何应对压力。与此

同时，在那之前的六个月，我非正式地表演了它。我认识的一位软件架构师说，在他被提升到这个职位18个月后，他明白了自己的职责是什么。这样的时间间隔是合理的，你需要知道你是否准备好朝着这个方向前进。即使你没有一个稳定的计划，最好开始采取一些小步骤来推动你前进，而不是停留在原地。

Standing in the same place in IT is a synonym for stagnation and personal fetters in life.

Translation: 站在IT的同一个位置，是生活停滞和个人羁绊的代名词。

Recommended Books

Translation: 推荐图书

Software Architecture in Practice (3rd Edition) (SEI Series in Software Engineering)

Translation: 软件体系结构实践（第三版）（软件工程SEI系列）

Essential Software Architecture

Translation: 基本软件体系结构